

Applicant : Bhanjois et al.
Serial No. : 09/408,149
Filed : September 29, 1999
Page : 2 of 9

Attorney's Docket No.: 07575-034001 / P01-1916.01

Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently amended) A computer program product ~~An operating system~~, tangibly stored on a machine readable medium, ~~the operating system~~ computer program product comprising:

a non-preemptive microkernel executing two or more processes in accordance with a non-preemptive scheduling scheme, wherein ~~[[each]]~~ a lower priority process executed by the non-preemptive microkernel relinquishes a processor for a higher priority process to execute only when the lower priority process blocks or explicitly requests to be preempted; and

one or more kernels each being executed as a process by the non-preemptive microkernel, wherein at least one of the one or more kernels executes an operating system as a dependent process, the operating system being a time-sliced operating system or a time-sliced microkernel.

2 - 3. (Cancelled)

4. (Currently amended) The computer program product ~~operating system~~ of claim 1, wherein the operating system is Unix.

5. (Currently amended) The computer program product ~~operating system~~ of claim 1, wherein each of the two or more processes executed by the non-preemptive microkernel has its own stack.

6. (Currently amended) The computer program product ~~operating system~~ of claim 1, wherein each of the two or more processes executed by the non-preemptive microkernel communicate using one or more messages.

Applicant : Bhanjois et al.
Serial No. : 09/408,149
Filed : September 29, 1999
Page : 3 of 9

Attorney's Docket No.: 07575-034001 / P01-1916.01

7. (Currently amended) The computer program product operating system of claim 1, wherein each of the two or more processes executed by the non-preemptive microkernel has a unique process identifier (PID).

8. (Currently amended) The computer program product operating system of claim 7, further comprising a mailbox coupled to a plurality of processes to service messages sent to a single PID.

9. (Currently amended) The computer program product operating system of claim 1, wherein each of the two or more processes executed by the non-preemptive microkernel never terminates.

10. (Currently amended) The computer program product operating system of claim 1, wherein one of the one or more kernels is a microkernel.

11. (Currently amended) A method for operating a computer system including a CPU, comprising:

managing two or more processes with a non-preemptive microkernel, the microkernel executing the two or more processes in accordance with a non-preemptive scheduling scheme, wherein [[each]] a lower priority process executed by the non-preemptive microkernel relinquishes the CPU for a higher priority process to execute only when the lower priority process blocks or explicitly requests to be preempted;

executing one or more kernels as one or more processes managed by the non-preemptive microkernel; and

executing an operating system in one of the one or more kernels as a dependent process, the operating system being a time-sliced operating system or a time-sliced microkernel.

12 - 13. (Cancelled)

Applicant : Bhanjois et al.
Serial No. : 09/408,149
Filed : September 29, 1999
Page : 4 of 9

Attorney's Docket No.: 07575-034001 / P01-1916.01

14. (Previously Presented) The method of claim 11, wherein the operating system is Unix.

15. (Previously Presented) The method of claim 11, wherein each of the two or more processes executed by the non-preemptive microkernel has its own stack.

16. (Original) The method of claim 11, further comprising performing inter-process communication using one or more messages.

17. (Previously Presented) The method of claim 11, wherein each of the two or more processes executed by the non-preemptive microkernel has a unique process identifier (PID).

18. (Previously Presented) The method of claim 17, further comprising servicing messages sent to a single PID by a plurality of processes using a mailbox.

19. (Previously Presented) The method of claim 11, further comprising executing the two or more processes without termination.

20. (Previously Presented) The method of claim 11, further comprising executing a microkernel in one of the one or more kernels.

Applicant : Bhanjois et al.
Serial No. : 09/408,149
Filed : September 29, 1999
Page : 5 of 9

Attorney's Docket No.: 07575-034001 / P01-1916.01

21. (Currently amended) A computer system, comprising:

a processor;

means for managing two or more processes with a non-preemptive microkernel, the microkernel executing the two or more processes in accordance with a non-preemptive scheduling scheme, wherein [[each]] a lower priority process executed by the non-preemptive microkernel relinquishes the processor for a higher priority process to execute only when the lower priority process blocks or explicitly requests to be preempted;

means for executing one or more kernels as one or more processes managed by the non-preemptive microkernel; and

means for executing an operating system in one of the one or more kernels as a dependent process, the operating system being a time-sliced operating system or a time-sliced microkernel.

22 – 23. (Cancelled)

24. (Previously Presented) The system of claim 21, wherein the operating system is Unix.

25. (Previously Presented) The system of claim 21, wherein each of the two or more processes executed by the non-preemptive microkernel has its own stack.

26. (Original) The system of claim 21, further comprising means for performing inter-process communication using one or more messages.

27. (Previously Presented) The system of claim 21, wherein each of the two or more processes executed by the non-preemptive microkernel has a unique process identifier (PID).

28. (Currently amended) The system of claim 27 [[21]], further comprising means for servicing messages sent to a single PID by a plurality of processes using a mailbox.

Applicant : Bhanjois et al.
Serial No. : 09/408,149
Filed : September 29, 1999
Page : 6 of 9

Attorney's Docket No.: 07575-034001 / P01-1916.01

29. (Previously Presented) The system of claim 21, further comprising means for executing each of the two or more processes executed by the non-preemptive microkernel without termination.

30. (Previously Presented) The system of claim 21, further comprising means for executing a microkernel in one of the one or more kernels.

31. (Currently amended) A computer, comprising:
an interconnect bus;
one or more processors coupled to the interconnect bus and adapted to be configured for server-specific functionalities including network processing, file processing, storage processing and application processing;
a configuration processor coupled to the interconnect bus and to the processors, the configuration processor dynamically assigning processor functionalities upon request;
one or more data storage devices coupled to the processors and managed by a file system;
a non-preemptive microkernel executing two or more processes in accordance with a non-preemptive scheduling scheme, wherein [[each]] a lower priority process executed by the non-preemptive microkernel relinquishes one of the one or more processors for a higher priority process to execute only when the lower priority process blocks or explicitly requests to be preempted; and
one or more kernels each being executed as a process by the non-preemptive microkernel, wherein at least one of the one or more kernels executes an operating system as a dependent process, the operating system being a time-sliced operating system or a time-sliced microkernel.

32. (Cancelled)

33. (Previously Presented) The computer of claim 31, wherein the non-preemptive microkernel executes a network switch operating system as a dependent process.